

Rechnerstrukturen und -organisation

Architektur- konzepte

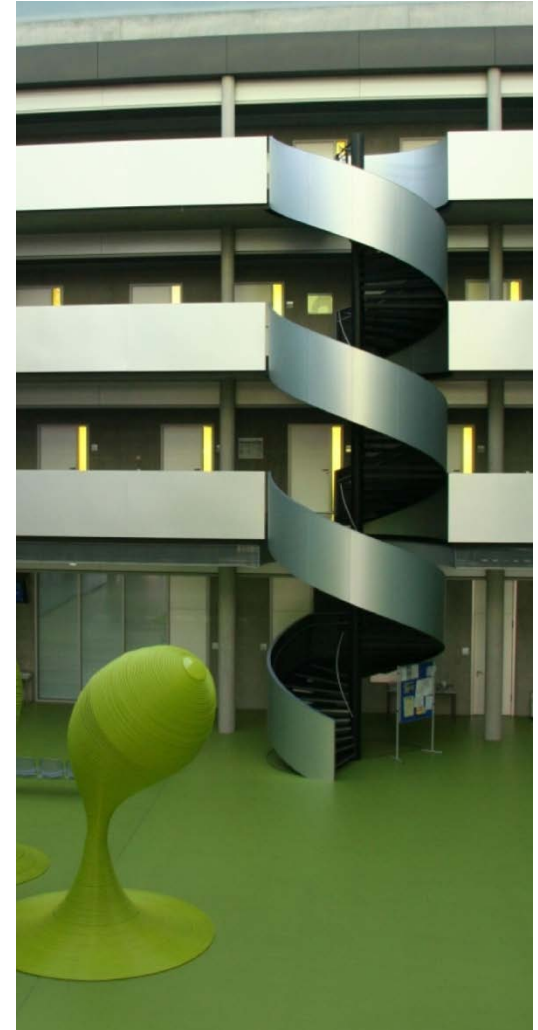
Rainer G. Spallek

TU Dresden, 08.12.2020



Gliederung

- 1 Zielstellung
- 2 Darstellung und Beschreibung eines Rechners
- 3 Schichtenmodell des Rechners
- 4 Hauptkomponenten des Rechners
- 5 Klassifikation von Rechnerarchitekturen
- 6 Rechnersysteme
- 7 Zusammenfassung



1 Zielstellung

- Übergang vom Schaltwerk zum Rechner.
- Darstellung und Komponenten von Rechnern.
- Modellierung von Rechnern.
- Erlangung eines Grundverständnisses für Rechnerarchitekturen.
- Auseinandersetzung mit den verschiedenen Architekturbegriffen.
- Einteilung, Klassifizierung von Rechnerarchitekturen.
- Vorstellung von Rechnersysteme und deren Beschreibung.

2 Darstellung und Beschreibung des Rechners

Rechner sind dem Prinzip nach nichts anderes als hochkomplexe, oft unüberschaubare, endliche Automaten, die wiederum durch hochkomplexe elektrische Schaltwerke realisiert sind.

Hauptbestandteile elektrischer Schaltwerke sind:

- Ein- und Ausgangsklemmen,
- Verknüpfungsglieder,
- Schaltnetze,
- Speicherglieder,
- Verbindungsnetzwerke

Die Anzahl der inneren Zustände eines Schaltwerkes ergibt sich direkt aus der Anzahl der darin vorhandenen Speicherglieder.

Vergleich Schaltwerk - Rechner

	„einfaches“ Schaltwerk	Rechner
Ein-, Ausgänge	endliche Anzahl	nahezu unendliche Anzahl
Schaltwerke	einfache Logik	komplex, verteilt
Speicherglieder	konzentriert angeordnet	über das System verteilt
innere Zustände	überschaubar, gering	unüberschaubar, extrem
Vernetzung	einfache Rückführungen	hochkomplex, lokal, global
Darstellung	logische Schaltungen	hochkomplexe Systeme
Beschreibung	Boolesche Gleichungen	Automaten-, Architekturmodelle

→ Rechner können aufgrund ihrer Komplexität nicht wie Schaltwerke dargestellt und beschrieben werden. Die Darstellung und Beschreibung erfolgt daher auf einem relativ hohem Abstraktionsniveau (Architekturebene).

Darstellungs- und Beschreibungsebenen

Ebene	Darstellung / Beschreibung
Architekturebene	System- und Verhaltensdarstellung Hardware-, Architekturbeschreibungssprachen, ...
Register-Transfer-Ebene (RTL)	Vernetzung von Komponenten, Modulen, Bussen Hardwarebeschreibungssprachen, ...
Logikebene	Vernetzung von Verknüpfungs- und Speichergliedern Netzlisten, Logikplan, ...
Schaltungsebene	Verdrahtung von Bauelementen sowie Realisierung der Herstellungstechnologie elektrische Netzwerke, Strom, Spannung, Ladung, ...

Einfache Rechnermodelle

Für die einfache Beschreibung und Modellierung von Rechnern sind spezielle, erweiterte Automatenmodelle erforderlich.

Turing – Maschine

Einfaches theoretisches Automatenmodell, das vor allem für theoretische Untersuchungen des Rechners gut geeignet ist:

Berechenbarkeit – Entscheidbarkeit – Akzeptierbarkeit

→ Zur Beschreibung praktischer, realer Rechner nicht gut geeignet.

Von – Neumann – Rechner

Erweitertes abstraktes Automatenmodell zur Beschreibung voll programmgesteuerter Rechner. Mit diversen Erweiterungen zur Beschreibung praktischer, realer Rechner geeignet.

→ Beschreibung auf Register-Transfer-Ebene (RTL) oder Architekturebene.

Rechner- und Prozessorarchitektur

Rechnerarchitektur

- Analyse und Synthese von Rechnerstrukturen und deren Organisation.
- Untersuchung des äußeren Erscheinungsbildes eines Rechners.
- Weitestgehende Abstraktion von inneren Vorgängen des Rechners.
- Klassifizieren, vergleichen, bewerten und entwerfen von Rechnern.
- Komponenten und deren Verbindungen in der Struktur von Rechnern.
- Kommunikations- und Verarbeitungsstruktur und deren Organisation.
- Implementierung und Verhältnis von Hard- und Softwarerealisierung.

Prozessorarchitektur

Beschränkung auf den Prozessor (auch Mikroprozessor, Mikrocontroller) als integrierte Verarbeitungseinheit und seine Verbindungen nach außen.

Grundaufgabe einer Rechneranlage ist das Sammeln, Speichern, Verarbeiten und Darstellen von Informationen.

Anwendungsgebiete von Rechneranlagen:

- **Universalrechner:** Auf jedem beliebigen Gebiet einsetzbar (flexibel).
- **Spezialrechner:** Nur für spezielle Aufgaben konzipiert (spezialisiert).

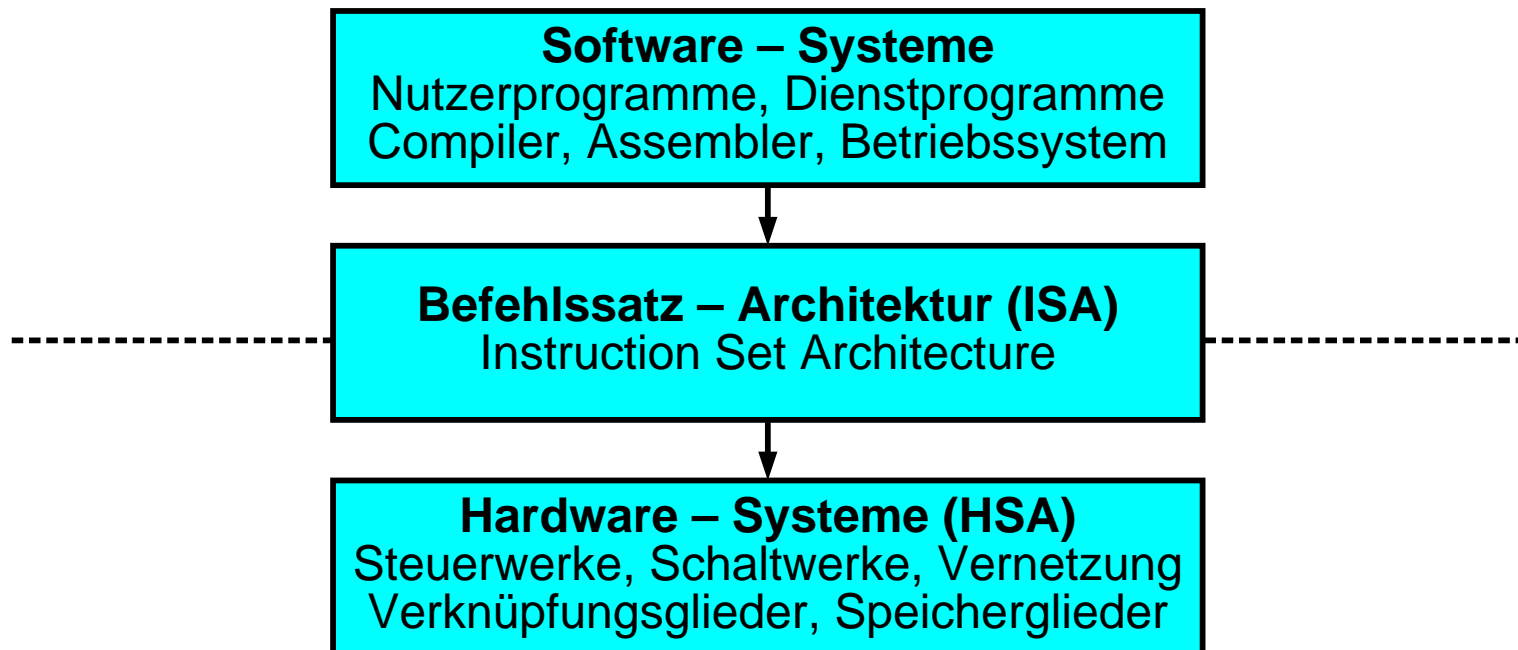
Forderungen an Computerarchitekturen:

- Rechenleistung, Echtzeitfähigkeit
- Erweiterbarkeit, Vernetzbarkeit
- Datendurchsatz, Latenz
- Ausfalltoleranz, Verfügbarkeit
- Bedienbarkeit, Benutzbarkeit
- Energieverbrauch

3 Schichtenmodell des Rechners



Vereinfachtes Schichtenmodell



Instruction Set Architecture (ISA):

Software-Hardware-Schnittstelle

Trennung von Programmiersprache, Compiler, Assembler, Betriebssystem auf der einen Seite und Befehlssatz, Adressierung, Maschinenprogramm, Speicherung auf der anderen Seite

Hardware System Architecture (HSA):

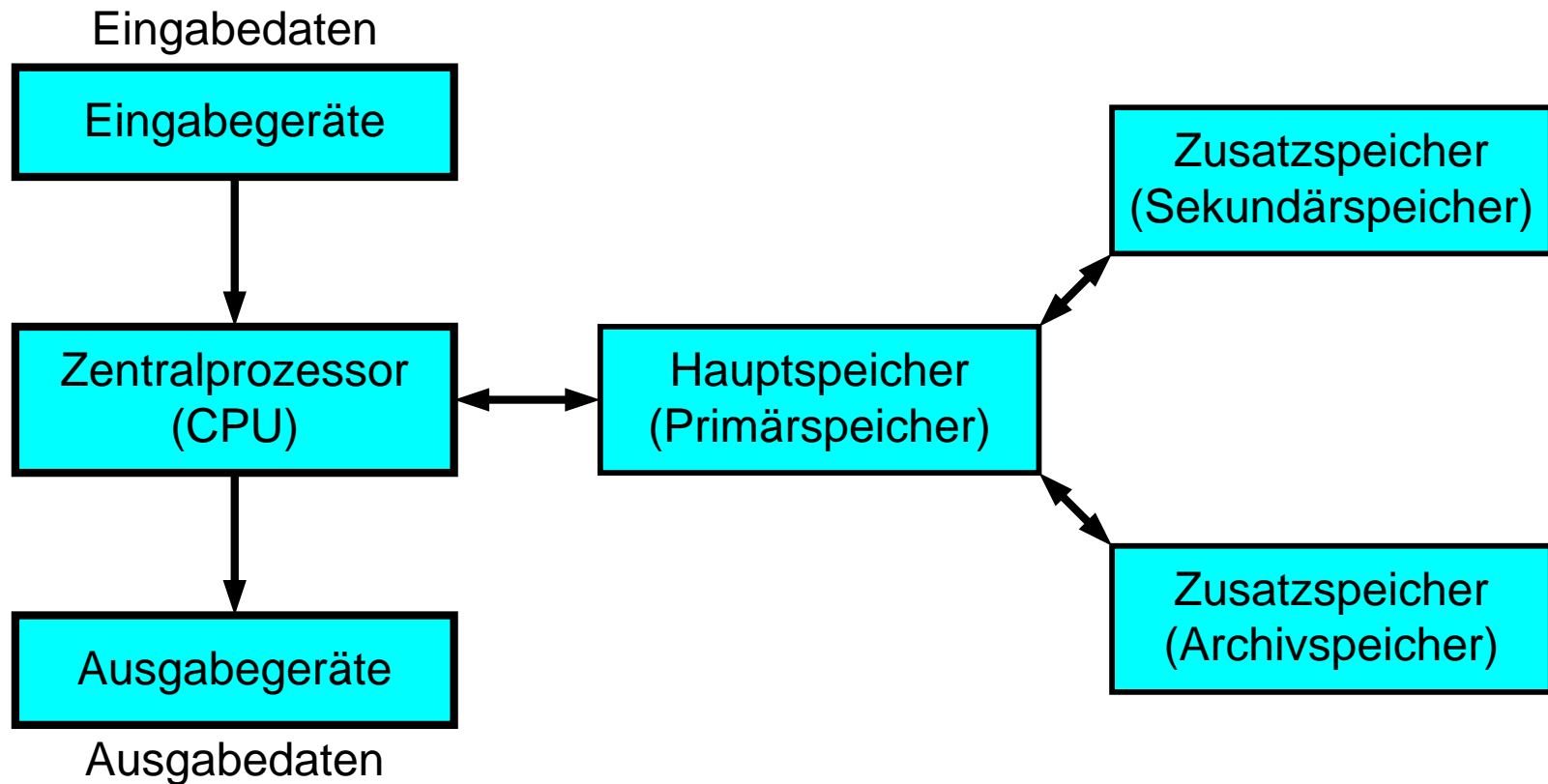
Physischer Aufbau der Architektur (Hardware).

Beschreibung des Organisations-, Struktur-, Operations-, und Kommunikationsprinzips. Definiert den Aufbau des Rechners aus den einzelnen Hardwarekomponenten und deren Vernetzung.

Makroarchitekturebene: Strukturelle Sicht → Struktur- und Organisation
→ Komponenten und Vernetzung.

Mikroarchitekturebene: Funktionale Sicht → Struktur und Organisation
→ Befehls- und Datenabarbeitung.

4 Hauptkomponenten des Rechners



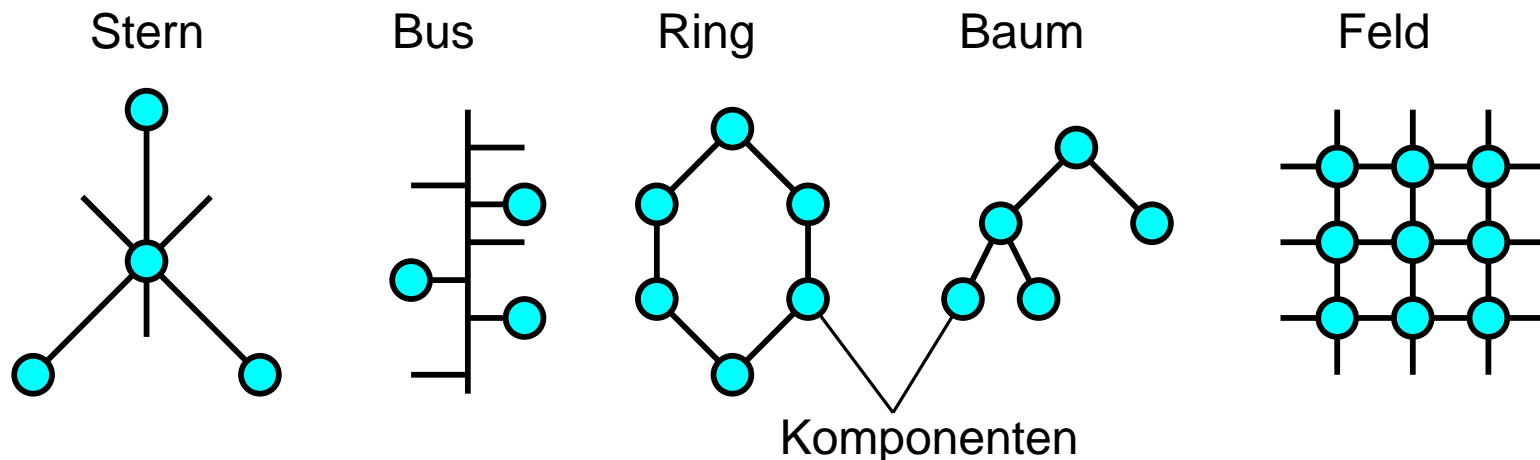
Hauptkomponenten

- CPU:** Zentrale Verarbeitungseinheit mit Rechenwerk und Steuerwerk (Mikroprozessor).
- Primärspeicher:** Hauptspeicher (DRAM, SRAM).
- Sekundärspeicher:** Zusatzspeicher mit direktem Zugriff (Festplatte).
- Archivspeicher:** Massenspeicher mit verzögertem Zugriff (CD, DVD, Tape).
- Systembus:** Verbindet alle Komponenten des Computers miteinander (PCI Bus, PCI-E Bus).
- Ein-/Ausgabegeräte:** Realisierung der Kommunikation des Computers nach außen (Monitor, Keyboard, Maus, Drucker, Netzkarte, Modem, Sound, . . .).

Verbindungsstruktur des Rechners

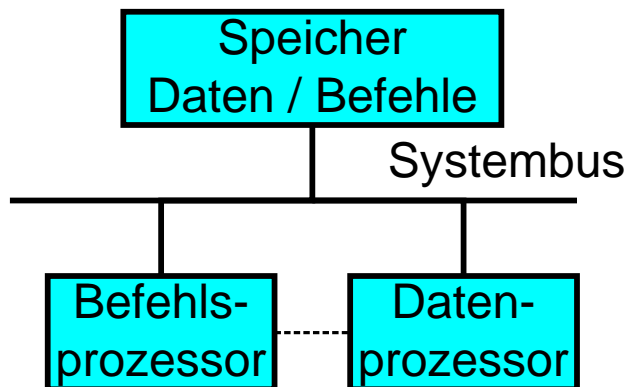
Die Verbindungsstruktur eines Rechners stellt ein Architekturmerkmal bzgl. der Verbindungen der einzelnen Rechnerkomponenten untereinander bzw. auch zwischen den Komponenten und der Umwelt dar (Ein-/Ausgabekomponenten).

Die Verbindungen können als Stern, Bus, Ring, Baum, Feld, Crossbar (steuerbarer Kreuzschienenverteiler), ... ausgeführt sein.

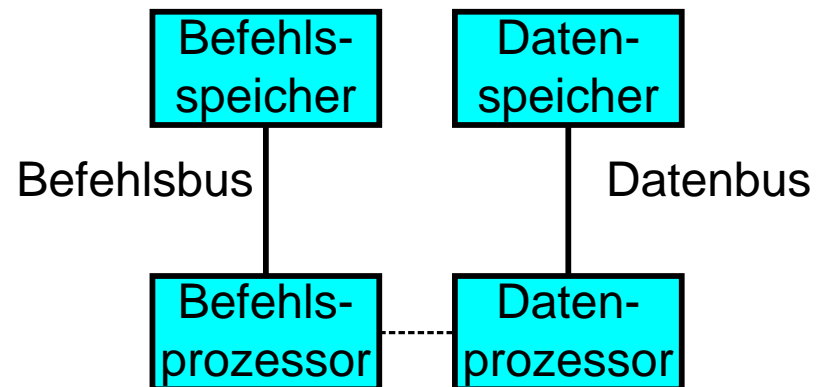


Busstrukturen

Princeton Architektur (John von Neumann)



Harvard Architektur (Howard Aiken)



Die Busse können unterteilt sein in Daten-, Adress-, Steuer- und Befehlsbus. Weitere Komponenten und abgesetzte Busse sind möglich.

5 Klassifikation von Rechnerarchitekturen

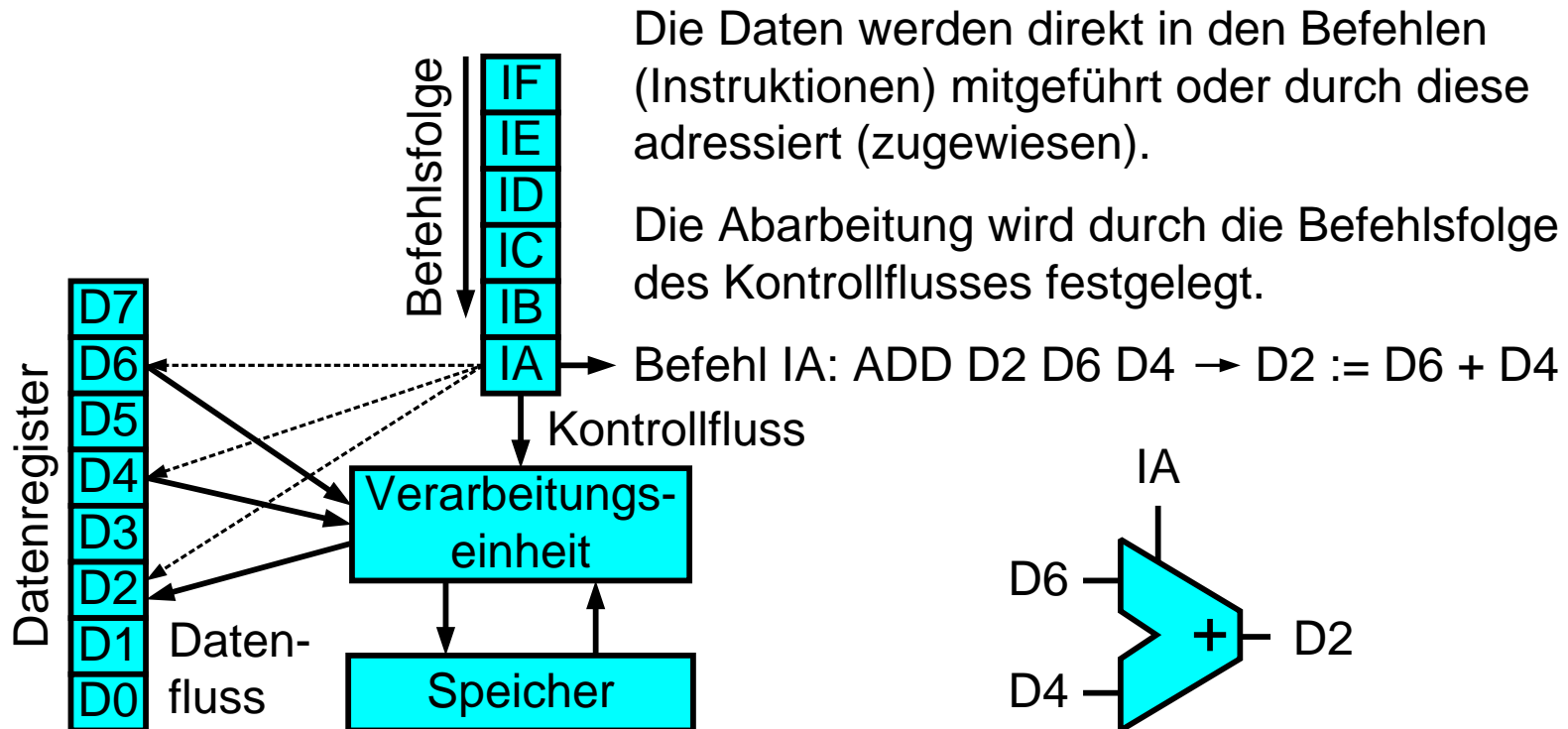
- Ordnungsstruktur und -prinzip von Rechnerarchitekturen.
- Anordnung, Organisation und Verbindungen der Rechnerkomponenten.
- Operations-, Verarbeitungs- und Kommunikationsprinzip des Rechners.
- Befehls- und Datenflüsse und deren Organisation.
- Parallelität auf den verschiedensten Ebenen.
- Prinzip der Datenspeicherung und -konsistenz des Rechners.

Eine eindeutige Klassifikation von Rechnerstrukturen ist nicht möglich. Vielmehr werden durch die Klasseneinteilung nur Grundprinzipien hervorgehoben. Die einzelnen Klassen können sich dabei weitgehend überdecken.

Die Zuordnung zu den einzelnen Klassen erfolgt jeweils nach der hervorzuhebenden Haupteigenschaft der Rechnerarchitektur.

Eine Architektur kann somit auch mehreren Klassen zugeordnet werden.

Kontrollflussarchitektur

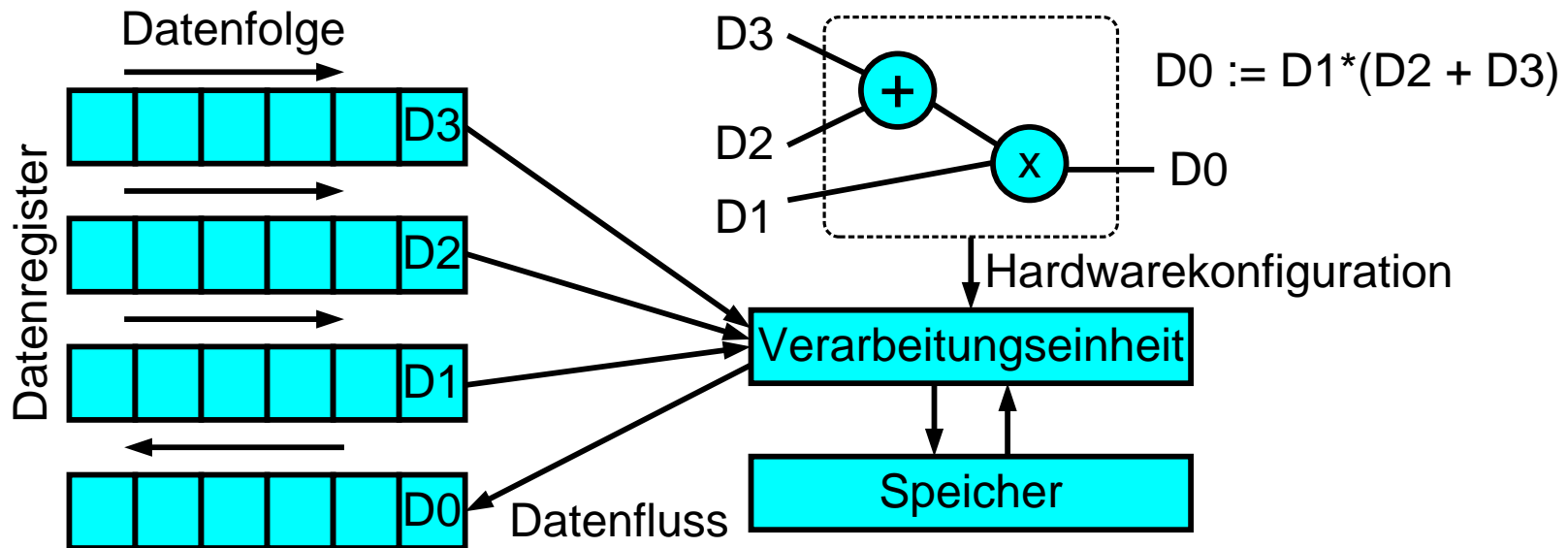


Kontroll- und Datenfluss sind orthogonal. Der Kontrollfluss steuert den Datenfluss.

Datenflussarchitektur

Die Befehle werden direkt im Datenfluss mitgeführt oder in der Verarbeitungseinheit durch den Datenfluss konfiguriert.

Die Abarbeitung wird durch den Datenfluss und die Hardwarekonfiguration bestimmt.



Der Datenfluss enthält implizit den Kontrollfluss, er bestimmt die Abarbeitung.

Kontrollflussarchitektur - Datenflussarchitektur

Kontrollflussarchitektur



Von-Neumann	RISC, CISC - Universalprozessor (GPP)
Harvard	ASIP - Applikationsspezifischer Befehlssatzprozessor
Superscalar	DSP - Digitaler Signalprozessor
VLIW	ASSP Applikationsspezifischer Signalprozessor
	RPA - rekonfigurierbare Computerarchitektur
	CCM - Custom Computing Machine
	Systolische Feld-Architekturen
	Xputer, Datenflussrechner

Datenflussarchitektur

Breites Spektrum von Architekturen zwischen Kontrollfluss- und Datenflussarchitektur.

Parallelitätsebenen in Rechnerarchitekturen

Bitenebene: BLP (Bit Level Parallelism)

Die Bitstellen eines Datums oder mehrere Daten werden parallel verarbeitet.

Befehlsebene: ILP (Instruction Level Parallelism)

Mehrere Befehle eines Kontrollflusses werden parallel ausgeführt.

Kontrollflussebene: MT (Thread Level Parallelism)

Parallele Ausführung mehrerer Teile (Threads) eines Kontrollflusses (Multithreaded).

Programmebene: MP (Application Level Parallelism)

Parallele Ausführung mehrerer Programme, Prozesse (Multiprocessing).

Datenflussebene: DFP (Data Flow Processing)

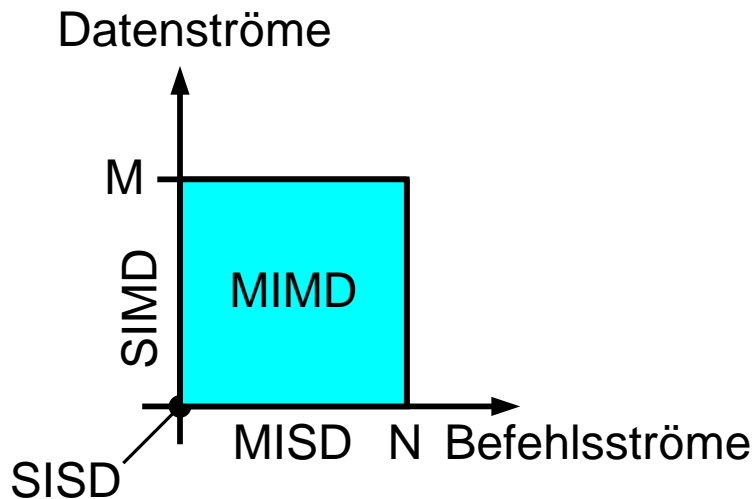
Die Datenflussverarbeitung ist vom Konzept her hochparallel.

Bit- und Befehlebenenparallelität werden feinkörnig (fine grained) bezeichnet, Kontrollfluss- und Prozessparallelität dagegen grobkörnig (coarse grained).

Parallelarchitekturen nach Flynn

Klassifikation nach M. Flynn (1972)

Zweidimensionale Klasseneinteilung der Rechnerarchitekturen nach der Anzahl der Befehls- und Datenströme.



		Datenströme	
		Single Data	Multiple Data
Befehlsströme	Single Instruction	SISD	SIMD
	Multiple Instruction	MISD	MIMD

Klasseneinteilung nach Flynn

SISD Single Instruction – Single Data (von-Neumann Rechnerkonzept)

MISD Multiple Instruction – Single Data, leere Klasse (Pipeline-Rechner ?)

SIMD Single Instruction – Multiple Data

Feldrechner: Parallele Anordnung gleichartiger Verarbeitungseinheiten, die alle gleichzeitig den selben Befehl ausführen.

Vektorrechner: Anordnung der Verarbeitungseinheiten als Pipeline zur zeitlich überlappten Befehlsabarbeitung (Vektor-, Matrixoperationen)

MIMD Multiple Instruction – Multiple Data, Multiprozessor und Multicomputer

SMT: Simultaneous Multithreaded

MTA: Multithreaded Architecture

MPP: Masively Parallel Multiprocessing

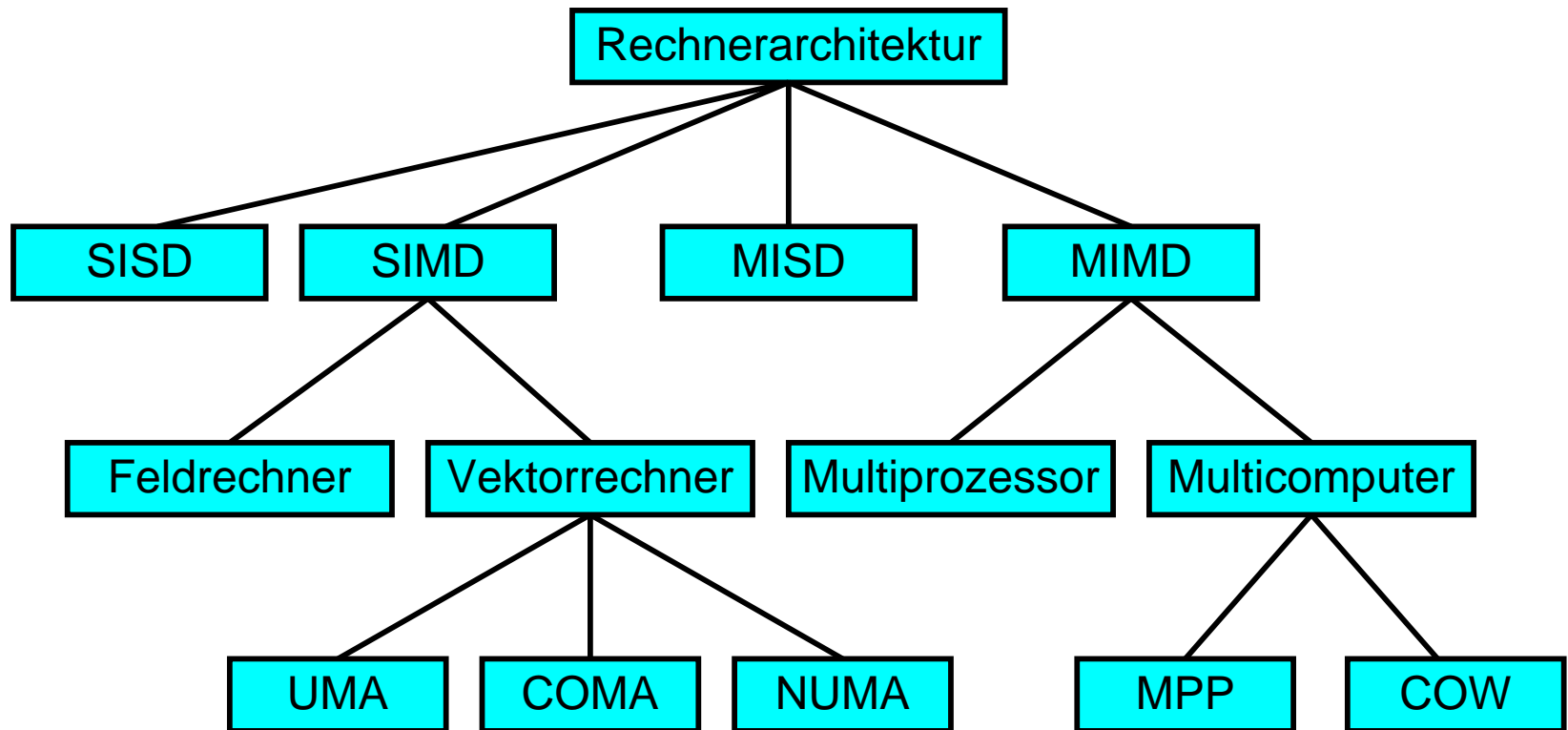
SMP: Symmetric Multiprocessing

COW: Cluster of Workstations

Speicheranbindung bei MIMD - Systemen

UMA	Uniform Memory Access, globaler gemeinsamer Adressraum
NUMA	Non Uniform Memory Access, verteilter gemeins. Adressraum
NORMA	No Remote Memory Access, physisch und programmtechnisch verteilter getrennter Speicher (z.B. Cluster)
COMA	Cache-Only Memory Access
CC-NUMA	Cache Coherent NUMA
NCC-NUMA	Non Cache Coherent NUMA
UCA	Uniform Communication Architecture
NUCA	Non Uniform Communication Architecture
DSM	Distributed Shared Memory
VSM	Virtual Shared Memory

Übersicht: Klassifikation nach Flynn



Klassifikation nach Verbindungsnetzwerken

Verbindungsnetzwerk:

statisch oder dynamisch konfigurierbar, rekonfigurierbar

Stern, Bus, Ring, Baum, Feld, Crossbar, . . .

Verbindungstopologie:

regulär, irregulär oder hierarchisch

Verbindungsaufbau:

zentral oder verteilt

Blockierungsverhalten:

blockierend, blockierungsfrei, oder rearrangierbar blockierungsfrei

Betriebsverhalten:

synchron oder asynchron

Vermittlungsart:

Leitungsvermittlung, Paketvermittlung oder Cut-Through-Routing

Klassifikation nach W. Giloi

Rechnerarchitektur: Element im kartesischen Produkt von Operationsprinzip und Struktur.

Operationsprinzip: funktionelles Verhalten der Architektur durch Festlegung einer Informationsstruktur und einer Kontrollstruktur.

Informationsstruktur: Lässt sich als Menge abstrakter Datentypen spezifizieren.

Rechnerstruktur: Gegeben durch die Art und Anzahl der Prozessoren, des Speichers, der Verbindungseinrichtungen und deren Kommunikationseinrichtungen.

Operationsprinzipien:

- von Neumann Operationsprinzip
- Operationsprinzipien der Programmparallelität
- Operationsprinzipien der Datenparallelität

Informationsstrukturen

Implizite Parallelität

Durch die Architektur gegebene Möglichkeiten, zur parallelen Abarbeitung.

Ebenen der Parallelität:

- Operationsebene (Arithmetische Operationen)
- Anweisungsebene (Ausführung von Anweisungen)
- Prozessebene
- Programmebene

Explizite Parallelität

Programm- oder auch Datenstrukturen erlauben explizit parallele Abarbeitung.

Unterschiedliche Klassen:

- standardisierten Programmstruktur
- standardisierten Datenstrukturen, Datenparallelität
- selbst-beschreibenden Informationseinheiten
- selbst-identifizierenden Daten

Kontrollstrukturen

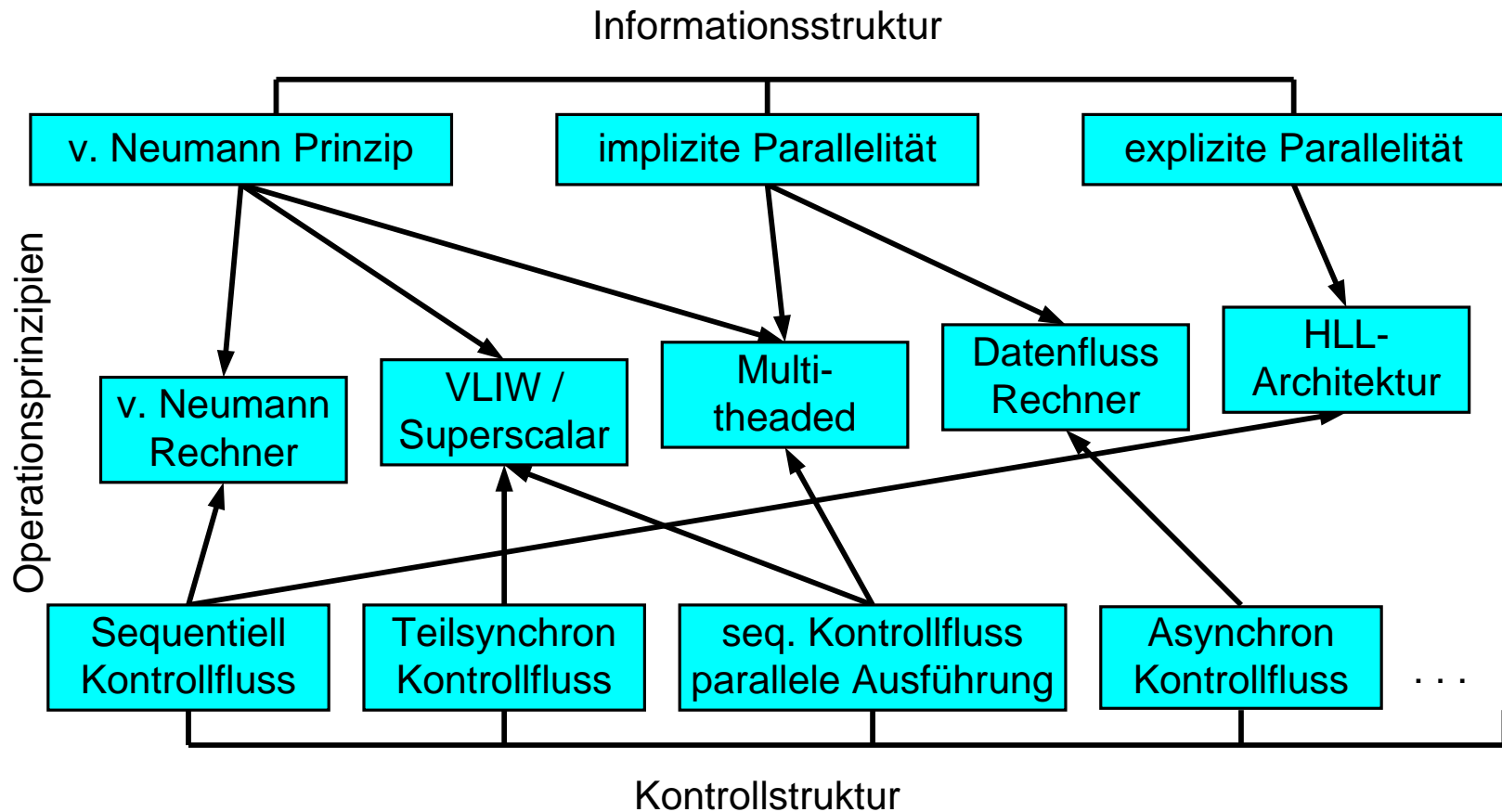
Die Kontrollstruktur eines Rechners wird durch die Spezifikation der Algorithmen für die Interpretation und Transformation der Informationskomponenten der Maschine bestimmt.

Folgende Kontrollstrukturen kann man beim streng sequentiellen Kontrollfluss des von Neumann Rechners unterscheiden:

- Konventioneller sequentieller Kontrollfluss für Programm und Daten.
- Sequentieller Kontrollfluss, aber gleichzeitig parallele Ausführung mehrerer Operationen in jedem Rechenschritt.
- Sequentieller Programmkontrollfluss bei assoziativem Zugriff auf Daten.
- Viele sequentielle Kontrollflüsse gleichzeitig.

Weitere Unterteilungen durch getrennte Betrachtung der Kontrolle des Programmablaufes und der Kontrolle des Datenzugriffes möglich.

Operationsprinzipien nach W. Giloi



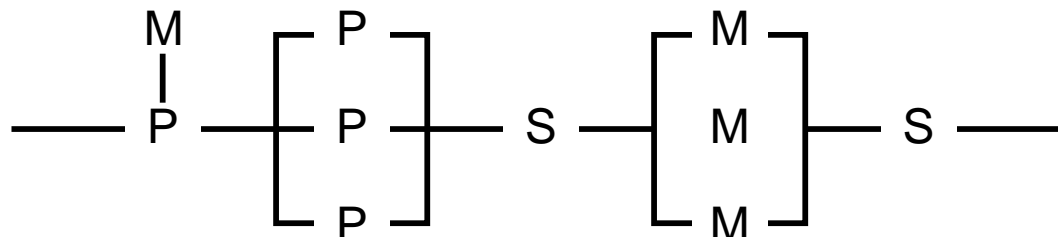
PMS Klassifikation (Processor, Memory, Switch)

Hardware-Beschreibungssprache für Rechnerarchitekturen

PMS-Symbole:

Symbol	Bezeichnung	Bemerkung
C	Computer	vollständige Rechnerbeschreibung
P	Prozessor	Prozessorkomponente
M	Memory	aufteilbarer adressierbarer Speicher
S	Switch	steuerbare Verbindungskomponenten (Bus, Crossbar, MUX)
...

PMS-Darstellung auch als Strukturdiagramm:



6 Rechnersysteme

Computertyp	Hauptanwendung
Wegwerfcomputer	Glückwunschkarten, Logistik, Identifikation
Controller	Eingebettete Systeme, Spezialanwendungen
Signalprozessor	Digitale Signalprozessoren, Kommunikationstechnik
Personalcomputer	Universalrechner, Arbeitsplatzrechner
Workstation, Server	Arbeitsplatzrechner, Netz-Server, Daten-Server
Workstation-Cluster	Abteilungsrechner, Hochleistungsrechner
Mainframe	Großrechner, Hintergrundrechner, Stapelverarbeitung
Supercomputer	Hochleistungsdatenverarbeitung, Wetterberechnung HPC (High Performance Computing)

7 Zusammenfassung

- Rechner sind aufgrund ihrer extrem hohen Komplexität nicht mehr wie einfache Schaltwerke darstellbar und beschreibbar.
- Für die Darstellung von Rechnern wird die Architekturebene bevorzugt.
- Die Beschreibung von Rechnern erfolgt weitestgehend abstrakt, z.B. durch Hardware-Beschreibungssprachen, Softwaremodellierung und -simulation.
- Rechner können zu Rechnersystemen zusammengefasst werden.
- Für die Steigerung der Verarbeitungsleistung von Rechnern wird Parallelität auf allen Ebenen bevorzugt.